

Listing of Claims

1. (Original) A method of setting up a procedure of a communication taking place between instances which is executable on a protocol tester of the type including the steps of selecting the instances involved in the communication, selecting a protocol layer on the basis of which the communication between the selected instances is to take place, selecting abstract communication interfaces of the protocol layer which are involved in the communication, selecting communication data, setting up a communication procedure executable between the instances through the protocol tester based on the several selecting steps, with the communication data selecting step being made graphically, with the parameters so selectable being allocated description files which are used in the setting up step, and with the possibility of selecting from a plurality of functionalities, each functionality being allocated a graphic representation and a description file, further comprising the steps of: creating a configuration file which is read in and interpreted at the time of the setting-up the communication procedure between the instances and from which is generatable at compile time an associated code; and entering information into the configuration file including a call name of an additional functionality in the executable code, a display form correlating with the additional functionality on a display on which it may be selected, and information on the description file which contains the executable source code of the additional functionality.
2. (Original) The method according claim 1 wherein for each parameter of the additional functionality to be entered and for the result of the additional functionality is also entered into the configuration file a name and a type.
3. (Original) The method according claims 1 or 2 wherein the display form is selected from the group consisting of a display name of the additional functionality and a graphic symbol for the additional functionality, the graphic symbol being allocated a graphic file in which the graphic symbol of the additional functionality is implemented.
4. (Original) The method according to claim 1 wherein the description file and/or the executable code are formulated in Forth, Jscript or VBScript.

5. (Original) The method according to claim 1 wherein the configuration file is implemented as a text file selected from the group consisting of an INI format and an XML format.
6. (Original) The method according to claim 1 wherein the additional functionality is entered in the configuration file.
7. (Original) The method according claim 6 wherein the configuration file further includes information on how many functionalities are stored in it.
8. (Original) The method according to claim 1 wherein for the implementation, into the executable source code of the additional functionality a call of the additional functionality is entered together with its call name.
9. (Original) The method according claim 8 wherein prior to the call the parameters required by the additional functionality are handed over, and after the call the result of the additional functionality is handed over.
10. (Original) The method according to claim 1 wherein reading-in of the description file occurs via an include command.
11. (Original) The method according to claim 1 wherein the instances involved in the communication are graphically selected, the protocol layer is graphically selected and/or the abstract communication interfaces of the protocol layer are graphically selected.
12. (Original) The method according to claim 1 wherein the abstract communication interfaces comprise SAPs (Service Access Points).
13. (Original) The method according to claim 1 wherein the communication data is selected from the group consisting of PDUs (Protocol Data Units) and ASPs (Abstract Service Primitives).

14. (Original) The method according to claim 1 wherein the communication data selecting step comprises the steps of: d1) graphically selecting a data format and; d2) graphically setting up a communication sequence between the instances involved.

15. (Original) The method according claim 14 wherein in step d2) the source code is enterable.

~~17.~~ 16. (Currently Amended) A protocol tester of the type having means for selecting the instances involved in the communication, means for selecting a protocol layer on the basis of which the communication between the selected instances is to take place, means for selecting abstract communication interfaces of the protocol layer which are involved in the communication, means for selecting communication data, and means for setting up a communication procedure executable between the instances through the protocol tester based on the several selecting means, with the communication data selecting means including graphic selection means with the parameters selectable by them being allocated description files which may be used by the setting-up means for setting up a communication procedure which is executable between the instances, and having the possibility of selecting from a plurality of functionalities, each functionality being allocated a graphic representation and a description file further comprising: means for creating a configuration file which may be read in and interpreted at the time of the setting-up of the communication procedure between the instances and from which is generatable at compile an associated code; and means for entering information into the configuration file including a call name of an additional functionality in the executable code, a display form correlating with the additional functionality on a display on which it may be selected, and information on the description file which contains the executable source code of the additional functionality.